JC978 U.S. PTO
10/015301
12/12/01

Bescheinigung   Certificate   Attestation

Die angehefteten Unterla-
gen stimmen mit der
ursprünglich eingereichten
Fassung der auf dem näch-
sten Blatt bezeichneten
europäischen Patentanmel-
dung überein.

The attached documents
are exact copies of the
European patent application
described on the following
page, as originally filed.

Les documents fixés à
cette attestation sont
conformes à la version
initialement déposée de
la demande de brevet
européen spécifiée à la
page suivante.

Patentanmeldung Nr.   Patent application No.   Demande de brevet n°

00127583.3

Der Präsident des Europäischen Patentamts;
Im Auftrag

For the President of the European Patent Office

Le Président de l'Office européen des brevets
p.o.

I.L.C. HATTEN-HECKMAN

DEN HAAG, DEN
THE HAGUE,   31/05/01
LA HAYE, LE

THIS PAGE BLANK (USPTO)

Europäisches
Patentamt

European
Patent Office

Office européen
des brevets

Blatt 2 der Bescheinigung
Sheet 2 of the certificate
Page 2 de l'attestation

Anmeldung Nr.:
Application no.:   00127583.3
Demande n°:

Anmeldetag:
Date of filing: 15/12/00
Date de dépôt:

Anmelder:
Applicant(s):
Demandeur(s):
International Business Machines Corporation
Armonk, NY 10504
UNITED STATES OF AMERICA

Bezeichnung der Erfindung:
Title of the invention:
Titre de l'invention:
   Method and system for automized and synchronous execution of customized code on off-loaded
   retrieved documents in a document processing system

In Anspruch genommene Prioriät(en) / Priority(ies) claimed / Priorité(s) revendiquée(s)

Staat: Tag: Aktenzeichen:
State: Date: File no.
Pays: Date: Numéro de dépôt:

Internationale Patentklassifikation:
International Patent classification:
Classification internationale des brevets:

/

Am Anmeldetag benannte Vertragstaaten:
Contracting states designated at date of filing: AT/BE/CH/CY/DE/DK/ES/FI/FR/GB/GR/IE/IT/LI/LU/MC/NL/PT/SE/TR
Etats contractants désignés lors du depôt:

Bemerkungen:
Remarks:
Remarques:

THIS PAGE BLANK (USPTO)

D E S C R I P T I O N

Method and System for Automized and Synchronous Execution of Customized Code on Off-loaded Retrieved Documents in a Document Processing System

Background of the Invention

The invention relates to document processing environments with
large document repositories and, more specifically, to a method
and system for processing off-loaded and/or retrieved documents
in a document processing.

Known client mailing applications like Lotus™ Notes™ or
Microsoft™ Outlook™ contain continuously growing document
repositories, namely the incoming and outgoing notes or emails
often including large attachments like text documents, graphics
or even storage consuming digitized pictures. Therefore, e.g. a
Lotus Notes application uses a Lotus Domino™ database from which
a tool like IBM Content Manager CommonStore™ for Lotus Domino
(CSLD) is used to move documents stored in that database to an
archive physically located on a different device like a tape
storage. CSLD thereupon allows to access documents that have
previously been archived.

CSLD also allows to access documents that have been archived
from any archive client application (e.g. scanning applications,
CommonStore for SAP™, etc). When documents are retrieved from
the archive to a Notes database, a Lotus Notes document is
created.

Lotus Notes supports writing of customized code which, for
instance, can be used to trigger workflow based on the state of
off-loaded documents.

The drawback of the existing approach is that customized code

can not be invoked synchronous to an archiving/off-loading and
retrieval process.

Summary of the Invention

It is therefore an object of the present invention to provide a
method and system as described beforehand which enable to invoke
customized code synchronous to an off-loading or retrieval
process.

It is another object to provide such a method and system
transparent to an end user without requiring any user
interaction.

The above objects are achieved by the features of the
independent claims. Advantageous embodiments are subject matter
of the subclaims.

The underlying idea is to provide a code component, in
particular a plug-in, that is automatically started before a
document is off-loaded, after it has been off-loaded or after it
has been retrieved.

The invention enables an advantageous pre-/post-processing of
documents in the above described document processing
environment. Invocation of at least one agent and execution of
customized code at a well-defined time, i.e. synchronous with
the underlying document processing step or event.

Brief Description of the Drawings

In the following, the present invention is described in more
detail by way of embodiments from which further features and
advantages of the invention become evident, where

Fig. 1   shows the various steps during archiving of a number
          of documents, invoking pre- and post-archiving agents
          in accordance with the invention; and

Fig. 2   shows an example of how pre- and post-archiving agents
          can be applied.

Detailed Description of the Drawings

In Fig. 1, a user selects 101 a number of documents for
archiving. The selection step creates 102 an archiving request,
in the present example, to IBM Content Manager CommonStore™ for
Lotus Domino (CSLD). CSLD is a tool to move Lotus Notes
documents in various formats to an archive.

CSLD prepares 103 the next document for archiving. If
configured, CSLD invokes 104 a pre-processing agent on the
current document. When the agent is finished, CSLD archives 105
the document. Then, if configured, CSLD invokes 106 a post-
processing agent on the current document. When the agent is
finished, CSLD marks 107 the document as 'archived'.

Fig. 2 illustrates how pre- and post-archiving agents can be
used to compute the amount of archived data and write it to a
database for accounting purposes.

It is noted that the dotted lines in the diagram separate the
main path of execution from the pre- and post-processing path.

A user selects 201 a number of documents for archiving. The
selection step creates 202 an archiving request, in the present
example, to CSLD. CSLD prepares 203 the next document for
archiving and invokes the pre-archiving agent on it. The code in
this agent computes 204 the current size of the document and the
number of attachments in the document. These values are written
205 to special fields in the current document.

When the agent has finished, CSLD archives 206 the document and invokes the post-archiving agent. The code in this agent removes 207 attachments and rich text items from the current document. After recalculating 208 the size of the document, the post-archiving agent logs 209 the size difference to an accounting database.

Finally, when the post-archiving agent is finished, CSLD marks 210 the document as 'archived'.

A similar procedure can be implemented using post-retrieval agents. In such a scenario, for instance, the amount of retrieved data can be written to an accounting database.

In CSLD, pre- and post-processing plug-ins are realized as the following agent invocation exits:

-   **Pre archiving agent:** A Notes agent that is invoked on a document right before it is archived. Usually used to prepare a document for archiving;

-   **Post archiving agent:** A Notes agent that is invoked on a document after it has been archived successfully. Can be used to delete the document, move it to a certain folder, create a stub document to release resources, collect accounting information, trigger workflows, etc.;

-   **Post retrieval agent:** A Notes agent that is invoked on a document after is has been retrieved from an archive. Can be used to set security properties, set workflow flags, write the document to a folder, etc.

All automatic agents are configured based on the Notes document form, i.e. for documents of different forms different agents can be configured. The agents are invoked synchronously. The current document being archived/retrieved is passed to the agent via the

session's document context. This allows to code the agent in
LotusScript or Java. Errors occuring during agent execution are
written to the CSLD trace file.

C L A I M S

1. A method for processing off-loaded and/or retrieved
    documents in a document processing system where requests
    are executed by an off-loading and retrieval process,
    comprising the steps of:

    Providing an interface to plugin customized code in order
    to execute the customized code;

    invoking customized code plugged into the interface when a
    request is issued thus being synchronous to the off-loading
    and/or retrieval process.

2. Method according to claim 1, wherein running the customized
    code synchronously to process documents before they are
    off-loaded, after they have been off-loaded, and/or after
    they have been retrieved.

3. Method according to claim 1 or 2, wherein providing a pre-
    archiving code or component being invoked on a document
    right before it is off-loaded.

4. Method according to any of claims 1 to 3, wherein providing
    a post-archiving code or component being invoked on a
    document after it has been off-loaded successfully.

5. Method according to any of the preceding claims, wherein
    providing a post-retrieval code or component being invoked
    on retrieved documents.

6. Method according to any of claims 3 to 5, wherein the code
    or component runs inside a plug-in architecture.

7. Method according to any of the preceding claims, wherein

    selecting a number of documents for archiving;

and wherein executing the following steps in a loop:

creating an archiving request to an archiving engine of the document processing system;

preparing a next document for archiving;

invoking an pre-archiving agent on the current document;

when the pre-archiving agent is finished, archiving the current document;

invoking the post-archiving agent on the current document; and

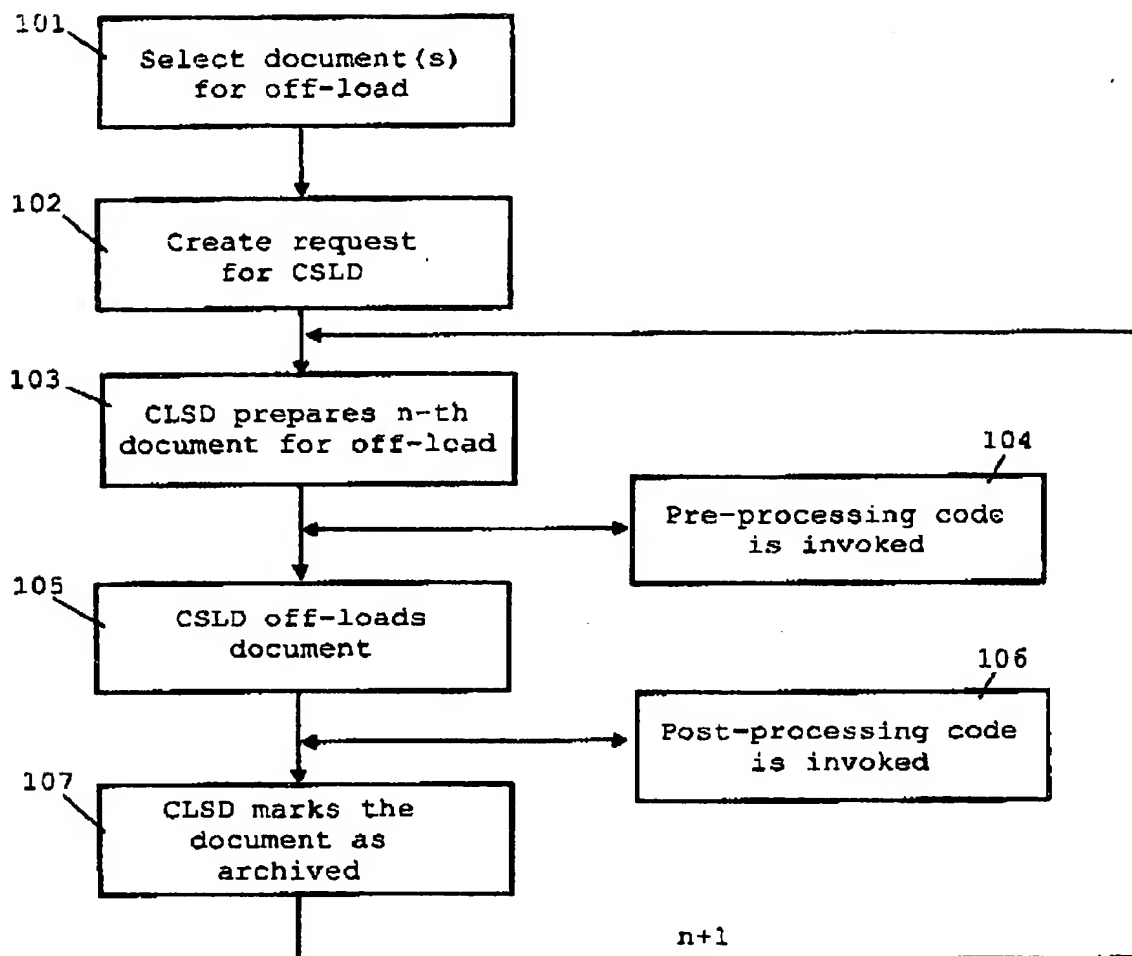when the post-archiving agent is finished, marking the document as 'archived'.

1 / 2

FIG. 1

Normal off-loading   Pre-processing   Post-processing
procedure

```
┌─────────────────────┐
│   Select document(s)│ ─201
│     for off-load    │
└─────────────────────┘
          │
          ▼
┌─────────────────────┐
│    Create request   │ ─202
│       for CSLD      │
└─────────────────────┘
          │
          ▼
┌─────────────────────┐
│   CLSD prepares with│ ─203
│     n-th document   │
└─────────────────────┘
```

Invoke

```
                    ┌─────────────────────┐
                    │   Calculate no of   │ ─204
                    │  attachments and size│
                    │     of document     │
                    └─────────────────────┘
                              │
                              ▼
                    ┌─────────────────────┐
                    │   Write results of  │
                    │ calculations back   │
                    │     to document     │
                    └─────────────────────┘ ─205
```

```
206
┌─────────────────────┐
│   Store document to │
│      repository     │
│  (incl. calculations)│
└─────────────────────┘
```

Invoke

```
                                        207
                              ┌─────────────────────┐
                              │  Remove attachments │
                              │  and richtext items │
                              │    from document    │
                              └─────────────────────┘
                                        │
                208                      ▼
                              ┌─────────────────────┐
                              │   Calculate size of │
                              │    document again   │
                              └─────────────────────┘
                                        │
                209                      ▼
                              ┌─────────────────────┐
                              │  Log size difference│
                              │     to accounting   │
                              │       database      │
                              └─────────────────────┘
```

```
210
┌─────────────────────┐
│   Mark document as  │
│       archived      │
└─────────────────────┘
```
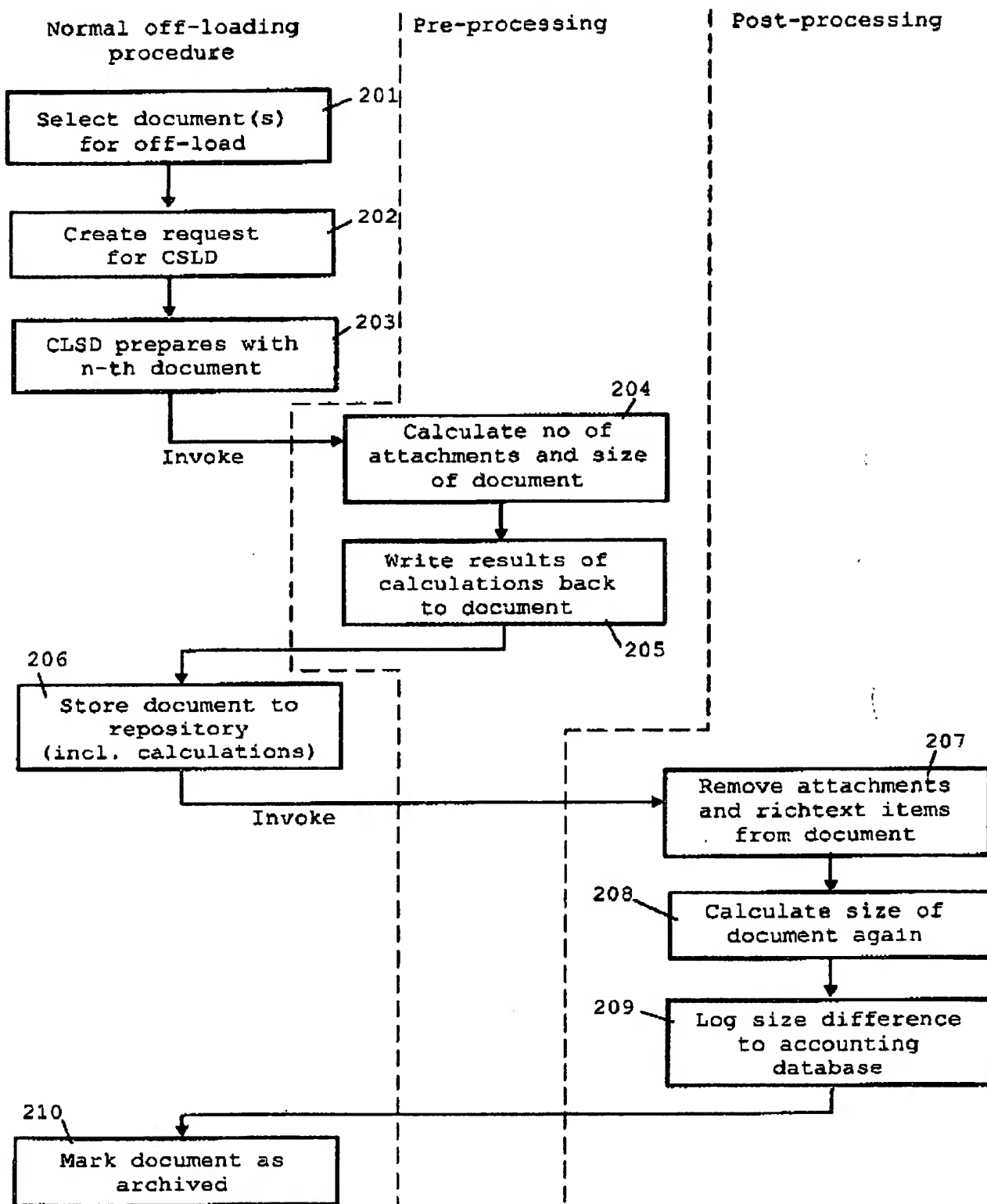
FIG. 2

A B S T R A C T

Disclosed are a method and system for executing customized code on off-loaded or retrieved documents in a document processing environment. The underlying idea is to provide a code component, in particular a plug-in, that is automatically started before a document is off-loaded, after it has been off-loaded or after it has been retrieved.

The invention enables an advantageous pre-/post-processing of documents in the above described document processing environment. Invocation of at least one agent and execution of customized code at a well-defined time, i.e. synchronous with the underlying document processing step or event. (Fig. 1)